

# Table of Contents

---

1. [Milestone 1](#)
2. [Milestone 2](#)
3. [Milestone 3](#)
4. [Milestone 4](#)

## Feedback | Group 6

---

### Milestone 1 | 2Oct-13Oct

1. **Define the problem:** done
  - Well defined!
2. **Finalizing roles:** done
3. **Create a product roadmap and prioritize functionality (items)** done
  - you did an excellent job, especially in MoSCoW method.
  - roadmap is realistic
4. **Creating the GitHub repository included readme.md and .gitignore (for Python) files:** done
5. Create a virtual environment in the above repo and generate requirements.txt (venv must be ignored in git) done
6. Push \*point 1, point 3, point 5 (requirements.txt). done
7. Complete the first chapter of Developing Python Packages completed by everyone
8. Create a private Slack channel in our Workspace and name it Group-{number} done
9. Schedule a call with me and Garo or come during office hours. done

**Grade:** 10/10 Good job!

# Milestone 2 | 16Oct-27Oct

---

## Fixes From the Milestone 1

Fixes were not required!

## Milestone 2

### Overall you did an excellent job!

I would recommend to move `raw_data` from package folder (`survival_analysis`). See, it is assumed that the package is going to be applied in real company. Where we have an existing database. Likewise moving `utils.py` one level higher, as in `utils` we are going to put functions which do not belong to any module.

#### 1. DB developer:

- Design the database using Star schema (provide ERD): **done**
- Insert Sample to data **done**

#### 2. Data Scientist:

- Complete data generation/acquisition/research: **done**
- Select data from DB: **done**
- Insert data to DB: **done**

#### 3. API developer:

- Select data from DB **done**
- Insert data to DB **done**
- Update data in DB **done**

#### 4. Finish the second chapter of Datacamp course **done by everyone**

#### 5. Finalize file/folder structure: relative imports must work properly **done, just the above mentioned movements**

- docs folder: putting all the documents there **done**
- models folder: putting modeling-related classes, functions **done**
- api folder: api related stuff **done**
- db folder: db related stuff **done**
- initialize `__init__.py` files accordingly (see Datacamp assignment chapter 1 and chapter 2) **done**
- logger folder: I will provide this module **done, try to use them in your py files**

*I can see multiple contributors!*

**Grade:** 20/20

## Milestone 3 | 30Oct-10Nov

---

1. Finish the **third** chapter of Datacamp course (please complete only the 3rd one)
2. **API Developer:**
  - Create a `run.py` file for an API (find the minimum workable example [here](#)). You have already done this
  - Test it on swagger You have already done this
  - following request types must be available to test (GET, POST, PUT), will provide more details on Friday. You have already done this
  - Think about endpoints which would top n% subs from output table ordered by Survival\_Rate (request such functionality from DB developer)
3. **DB developer:** You have already done this, complete all the methods
  - Create all the functionalities that your are going to need from SQL side (discuss with Product Manager, share it with API developpe) done
  - complete/fix the methods from `SQLHandler()` class
  - finalize the documentation for `schema.py` by using `pyment` package done
  - finalize the documentation for `SQLHandler()` by using `pyment` package done
4. **Data Scientist:** start working on modeling part, by selecting the date from SQL DB
  - we just need to run sample model and store the output to sql pdone
5. **Product Manager** done
  - since you have partially done 2-3 points, concentrate on the application scenario (at least two)

**Grade:** 30/30 Good job!

## Milestone 4 | 26 Nov-6 Dec

1. Complete the Datacamp course
2. Create an `example.ipynb` file and implement all the functionality of the package (make shure to make do it chunk by chunk, in order to convert it `reveal.js presentation). This is going to be part of the demo
3. As soon as you finish the documentation us `Mkdocs` in order to generate docs.html file, which is going to be hosted on GitHub
4. publish you package to `pypi.org`